



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **07129407 A**

(43) Date of publication of application: 19 . 05 . 95

(51) Int. Cl.

**G06F 9/445****G06F 9/06****G06F 13/00**(21) Application number: **05276269**

(22) Date of filing: 05 . 11 . 93

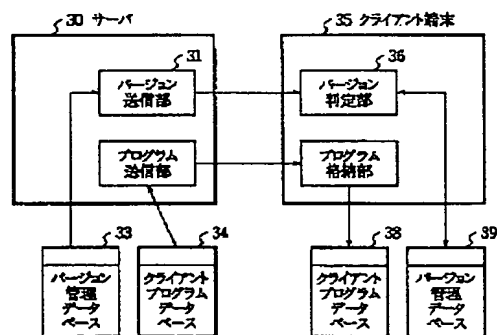
(71) Applicant: **NIPPON DENKI JOHO SERVICE  
KK**(72) Inventor: **ENDO TSUNEO**(54) **VERSION MANAGING METHOD**

## (57) Abstract:

**PURPOSE:** To reduce man-hour and cost for version up by performing update processing for transferring the program of the latest version from a server and a client terminal and for storing it in a program data base.

**CONSTITUTION:** A server 30 reads version information from a version managing data base 33 and transmits it to a client terminal 35. A version decision part 36 at the client terminal 35 compares that version information with version information stored in a version managing data base 39 and decides whether a program preserved at the client terminal 35 is updated or not when the information is not matched and when it is judged not to update that program, conventional processing is performed. When it is judged to update the program, the update processing is performed to transfer the program of the latest version from the server 30 to the client terminal 35 and to store it in a program data base 38 of the client terminal 35.

COPYRIGHT: (C)1995,JPO



[0004]

[Problems to be Solved by the Invention]

The method of maintaining system programs in a computer system such as the above-described conventional server-client type system entails problems described below. When system programs are upgraded, or when a bug correction, a revision or the like is made, it is impossible to collectively install upgraded, corrected or revised programs on the vendor side and there is a need to perform upgrading operations at user's places to which the computer system has been provided. In particular, in a case where many, e.g., several hundred users using the system program are spread throughout a country, a huge number of operation steps is required and the total cost of upgrading is accordingly high. Also, in the case where the installing work is performed by SEs on the user side, the skills of user SEs may vary, which may present a problem in terms of reliability.

[0005]

[Means for Solving the Problems]

A version management method of the present invention comprises performing circuit processing for connecting a server and a client terminal, making a determination on the

basis of a user identifier and a password input by a user as to whether the user has been authorized to use the network, making a determination as to whether a program version in system version information from the server is equal to a program version held by the client terminal when the system version information is transmitted from the server to the client terminal, performing ordinary processing corresponding to some of various menu items, e.g., mail or an electronic notice board if the program version in the system version information is equal to the program version held by the client terminal, making a decision to update or not to update the program held by the client terminal if the program version in the system version information is not equal to the program version held by the client terminal, performing the ordinary processing when a decision is made not to update the program held by the client terminal, and performing updating processing when a decision is made to update the program held by the client terminal, the updating processing comprising transmitting the latest version of the program from the server to the client terminal and storing the latest version of the program in a program data base of the client terminal.

[0006]

[Embodiment]

An embodiment of the present invention will now be

described with reference to the drawings.

[0007]

Fig. 1 is a flowchart for explaining an embodiment of the present invention, Fig. 2 is a block diagram showing the relationship between a server and client terminals in the embodiment shown in Fig. 1, and Fig. 3 is a diagram showing the contents of updating processing in the embodiment shown in Fig. 1.

[0008]

In the embodiment shown in Fig. 1, circuit processing for connecting a server 30 and a client terminal 35 (see Fig. 2) is first performed in step 10. Then, in step 11, a user identifier (identification code: ID) and a password (PW) input by a user are checked. If the input user identifier (identification code: ID) and password (PW) are authorized ones, the process proceeds to step 12 and version determination processing is performed.

[0009]

In step 12, the server 30 reads out version information from a version management data base 33, and the version information is transmitted from a version transmitting section 31 to the client terminal 35. A version determination section 36 in the client terminal 35 compares the received version information with version information stored in a version

management data base 39. If it is thereby determined that the version information stored in the version management data base 39 is the latest one, the process proceeds to step 14 and ordinary processing is performed. If it is determined that the version information stored in the version management data base 39 is an older one, the process proceeds to step 13. As ordinary processing in step 14, processing corresponding to some of various menu items, e.g., electronic mail, or an electronic notice board is performed.

[0010]

In step 13, the user is notified that the version information presently stored in the version management data base 39 of the client terminal 35 is out-of-date, thereby being urged to make a decision of subsequent processing. This notification to the user is performed every time if the version information sent from the server 30 differs from the version information stored in the version management data base 39 of the client terminal 35, and is therefore effective in preventing omission of updating. The decision of subsequent processing is either a one to continue processing without version updating or the one to perform version updating. In the case of a decision not to perform version updating, the process proceeds to step 14. In the case of a decision to perform version updating, the process proceeds to step 15.

[0011]

Updating processing in step 15 has three updating modes shown below, and is performed according to the usage conditions by the user.

[0012]

- (1) A mode in which all programs are updating targets,
- (2) a mode in which updating to only new programs is performed according to version information from the client terminal 35, and
- (3) a mode in which the user updates only a specific program.

Proper usage of these modes is as described below. For example, in a case where almost all programs are updating targets, a case where a transfer abnormality has occurred during updating, or a case where it is desirable to collectively store programs in a different directory, the mode described in (1) is used. In a case where it is desirable to complete updating of the desired program in a short time, the mode described in (2) is used. In a case where a particular program is destroyed or erased for some reason, the mode described in (3) is used.

[0013]

Processing in the mode described in (1) is performed as described below. A list of all programs is transmitted from the server 30 to the client terminal 35. The client terminal

35 successively sends requests for transmission of programs referred to in order in the list. The programs received from the server 30 in correspondence with the requests are stored in a client program data base 38. After the completion of storage of the programs for which the requests have been sent, the contents of a client-side version management table 55a stored in the version management data base 39 of the client terminal 35 are replaced by rewriting with the contents of a server-side version management table 51 stored in the version management data base 33 of the server 30, thus updating the client-side version management table 55a.

[0014]

Processing in the mode described in (2) is performed as described below. Updating date information is transmitted from the client terminal 35 to the server 30. The server selects programs updated later relative to the updating dates in the information and sends a list of these programs to the client terminal 35. The client terminal 35 successively sends requests for transmission of programs referred to in order in the list, and stores in the client program data base 38 the programs received from the server 30 in correspondence with the requests. After the completion of storage of the programs for which the requests have been sent, updating of the client-side version management table is performed in the same manner

as that in the mode described in (1).

[0015]

Processing in the mode described in (3) is performed as described below. A request for transmission of a particular program is transmitted from the client terminal 35 to the server 30. The program is received and stored in the client program data base 38. In this case, updating of the client-side version management table is not performed.

[0016]

After the completion of updating processing in step 15, the process proceeds to step 16 and verification determination is performed.

[0017]

Verification determination in step 16 is performed by outputting a message for urging the user to choose whether or not to perform the operation for examining whether the program that has undergone updating processing in step 15 is normal. If a decision is made to perform verification, the process proceeds to step 17. If a decision is made not to perform verification, the process proceeds to step 18 by skipping over step 17.

[0018]

In step 17, the program that has undergone updating processing in step 15 is again received from the server 30 and



is compared with the corresponding program stored in the client program data base 38 by the preceding processing. This operation is performed on the basis of the list used in step 15. After the completion of processing in step 17, the process proceeds to step 18.

[0019]

In step 18, a message is output to urge the user to choose whether or not to perform the ordinary processing in accordance with the updated new program. If a decision is made to perform the ordinary processing, the process proceeds to step 14. If a decision is made not to perform the ordinary processing, the process proceeds to step 19 to end.

[0020]

In step 19, circuit processing for cutting the connection between the server 30 and the client terminal 35 is performed, thereby terminating the sequence of processing steps.

特開平 7-129407

(43) 公開日 平成7年(1995)5月19日

(51) Int. Cl.<sup>6</sup>

識別記号

庁内整理番号

F I

技術表示箇所

G 0 6 F 9/445

9/06

4 1 0 Q

9367-5 B

13/00

3 5 7 Z

7368-5 B

9367-5 B

G 0 6 F 9/06 4 2 0 M

審査請求 未請求 請求項の数 1

O L

(全 4 頁)

(21) 出願番号 特願平5-276269

(22) 出願日 平成5年(1993)11月5日

(71) 出願人 390001041

日本電気情報サービス株式会社

東京都港区三田1丁目4番28号

(72) 発明者 遠藤 恒夫

東京都港区三田一丁目4番28号 日本電気

情報サービス株式会社内

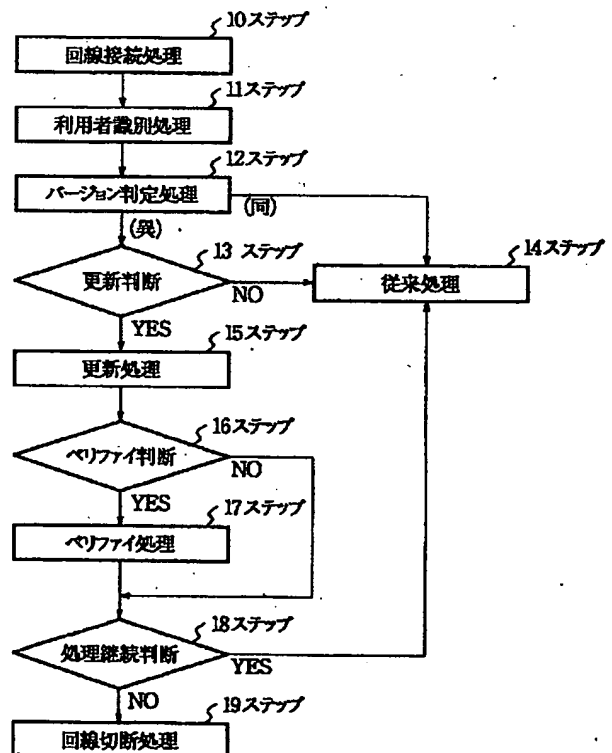
(74) 代理人 弁理士 京本 直樹 (外2名)

(54) 【発明の名称】 バージョン管理方法

(57) 【要約】

【構成】 サーバからクライアント端末に対してシステムバージョン情報を転送し、システムバージョン情報中のプログラムバージョンがクライアント端末が保有するプログラムバージョンとが一致するか否かの判定し、サーバのプログラムバージョンがクライアント端末のプログラムバージョンと一致しないとき、クライアント端末のプログラムを更新するか否かの判断を行い、更新すると判断した場合はサーバからクライアント端末に対して最新バージョンのプログラムを転送してクライアント端末のプログラムデータベースに格納する更新処理を行う。

【効果】 従来人手で行っていたシステムプログラムのバージョンアップの作業を、自動的に行うことが可能になり、従ってバージョンアップのための作業工数と費用とを節減できる。また、ユーザの S E にインストールして貰う必要もなくなるため、信頼性を向上できる。



## 【特許請求の範囲】

【請求項1】 サーバとクライアント端末とを接続する回線処理を行い、ユーザ識別子とパスワードとを入力して利用者が当該ネットワークを利用できる資格を有するか否かを判別し、前記サーバから前記クライアント端末に対してシステムバージョン情報を転送したとき、前記システムバージョン情報中のプログラムバージョンが前記クライアント端末が保有するプログラムバージョンとが一致するか否かの判定し、前記システムバージョン情報中のプログラムバージョンが前記クライアント端末が保有するプログラムバージョンと一致するときには、メールや電子掲示板等の各種メニューに従って処理を行う従来处理を行い、前記システムバージョン情報中のプログラムバージョンが前記クライアント端末が保有するプログラムバージョンと一致しないときには、前記クライアント端末が保有する前記プログラムを更新するか否かの判断を行い、前記クライアント端末が保有する前記プログラムを更新しないと判断した場合は前記従来处理を行い、前記クライアント端末が保有する前記プログラムを更新すると判断した場合は、前記サーバから前記クライアント端末に対して最新バージョンのプログラムを転送して前記クライアント端末のプログラムデータベースに格納する更新処理を行うことを含むことを特徴とするバージョン管理方法。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、ネットワークを用いたコンピュータシステムにおいて、ネットワークを介してプログラムを配信することによって端末装置のシステムバージョンを管理するためのバージョン管理方法に関する。

## 【0002】

【従来の技術】 近年のコンピュータシステムは、社会の要請に従ってダウンサイジング化が普及の一途を辿っている。メインフレームを中心とする従来のコンピュータシステムにおいては、ソフトウェアのバージョンアップ（機能追加や機能変更等のプログラムの更新）は、ホストコンピュータのプログラムの更新だけで充分であったが、コンピュータシステムがサーバとクライアント端末とで構成されるいわゆるサーバ・クライアント型システムになると、サーバおよびクライアント端末の両者にシステムプログラムが必要となり、それらのメンテナンスについて十分に配慮する必要が生じてきている。

【0003】 このため、システムプログラムのインストールやバージョンアップの作業は、ソフトウェアの出荷元において、ハードウェアに直接にインストールしてユーザに出荷するか、または、エンジニアがシステムプログラムを格納したフロッピディスクまたはデータカー

トリッジをユーザの所に持参し、ユーザの所でインストールするか、または、ユーザに専門のSEがいるとき、システムプログラムを格納した媒体をユーザに送付してユーザのSEにインストールして貰うか何れかの方法によって行っている。

## 【0004】

【発明が解決しようとする課題】 上述したような従来のサーバ・クライアント型システムのコンピュータシステムにおけるシステムプログラムのメンテナンス方法は、システムプログラムのバージョンアップがあったときやバグ修正・改訂等があったときは、出荷元において一括してインストールすることが不可能なため、コンピュータシステムの出荷先であるユーザの所でシステムプログラムのバージョンアップの作業を行わなければならない、特に数百という多数のユーザが全国的に展開されているような場合は、膨大な作業工数がかかり、またそのための費用も大きいという問題点を有している。また、ユーザのSEにインストールして貰う場合は、ユーザのSEの技術力にばらつきがあるため、信頼性の点で不安があるという問題がある。

## 【0005】

【課題を解決するための手段】 本発明のバージョン管理方法は、サーバとクライアント端末とを接続する回線処理を行い、ユーザ識別子とパスワードとを入力して利用者が当該ネットワークを利用できる資格を有するか否かを判別し、前記サーバから前記クライアント端末に対してシステムバージョン情報を転送したとき、前記システムバージョン情報中のプログラムバージョンが前記クライアント端末が保有するプログラムバージョンとが一致するか否かの判定し、前記システムバージョン情報中のプログラムバージョンが前記クライアント端末が保有するプログラムバージョンと一致するときには、メールや電子掲示板等の各種メニューに従って処理を行う従来处理を行い、前記システムバージョン情報中のプログラムバージョンが前記クライアント端末が保有するプログラムバージョンと一致しないときには、前記クライアント端末が保有する前記プログラムを更新するか否かの判断を行い、前記クライアント端末が保有する前記プログラムを更新しないと判断した場合は前記従来处理を行い、前記クライアント端末が保有する前記プログラムを更新すると判断した場合は、前記サーバから前記クライアント端末に対して最新バージョンのプログラムを転送して前記クライアント端末のプログラムデータベースに格納する更新処理を行うことを含むものである。

## 【0006】

【実施例】 次に、本発明の実施例について図面を参照して説明する。

【0007】 図1は本発明の一実施例を示すフローチャート図、図2は図1の実施例におけるサーバとクライアント端末との関係を示すブロック図、図3は図1の実施

例における更新処理の内容を示す説明図である。

【0008】図1の実施例は、まずステップ10においてサーバ30とクライアント端末35とを接続する(図2参照)回線処理を行った後、ステップ11において、利用者が入力したユーザ識別子(識別コード:ID)とパスワード(PW)とをチェックする。入力したユーザ識別子(識別コード:ID)およびパスワード(PW)が正規のものである場合は、ステップ12に移行してバージョン判定処理を行う。

【0009】ステップ12においては、図2に示すように、サーバ30がバージョン管理データベース33からバージョン情報を読み出し、それをバージョン送信部31からクライアント端末35に対して送信する。クライアント端末35のバージョン判定部36は、送られてきたバージョン情報をバージョン管理データベース39に格納してあるバージョン情報と比較し、バージョン管理データベース39に格納してあるバージョン情報が最新のものであると判断したときは、ステップ14に移行して従来処理を行い、バージョン管理データベース39に格納してあるバージョン情報が古いものであると判断したときは、ステップ13に移行する。ステップ14における従来処理は、メールや電子掲示板等の各種メニューに従った処理を行う。

【0010】ステップ13においては、現在クライアント端末35のバージョン管理データベース39に格納してあるバージョン情報が古いものであることをユーザに通知し、その後の処理に対する判断を促す。この通知は、サーバ30から送られてきたバージョン情報がクライアント端末35のバージョン管理データベース39に格納してあるバージョン情報と異なる場合は、毎度ユーザに対して行われるため、更新忘れを防止する効果がある。後処理に対する判断は、バージョンの更新は行わずにそのまま処理を続行するという判断と、バージョンの更新を行うという判断との何れか一方である。バージョンの更新を行わないという判断のときはステップ14に移行し、バージョンの更新を行うという判断のときはステップ15に移行する。

【0011】ステップ15における更新処理は、次の三つの更新モードがあり、ユーザの使用状況に応じて処理される。

【0012】(1) すべてのプログラムを対象として更新を行うモード

(2) クライアント端末35のバージョン情報によって新しいプログラムのみを更新を行うモード

(3) ユーザが特定のプログラムのみを更新するモード

これらのモードの使い分けは、例えば、大部分のプログラムが更新対象となっている場合や、更新の途中で転送に異常が発生した場合や、別のディレクトリにまとめて格納したい場合は(1)のモードを使用する。短時間に

目的のプログラムの更新を終了したい場合は(2)のモードを使用する。運用中に何らかの理由によって特定のプログラムが破壊された場合や消去された場合は、(3)のモードを使用する。

【0013】(1)のモードによる処理は、サーバ30からクライアント端末35に対してすべてのプログラムの一覧表を送信し、クライアント端末35は、その一覧表から順番にプログラムの送信要求を出し、これによってサーバ30から受信したプログラムをクライアントプログラムデータベース38に格納する。要求をしたプログラムの格納が終了すると、図3に示すように、クライアント端末35のバージョン管理データベース39に格納してあるクライアント側バージョン管理テーブル55aの内容を、サーバ30のバージョン管理データベース33に格納してあるサーバ側バージョン管理テーブル51の内容に書換え、クライアント側バージョン管理テーブル55bを更新する。

【0014】(2)のモードによる処理は、クライアント端末35からサーバ30に対して更新日時情報を送信し、サーバ30は、それを基にその情報よりも新しいプログラムのみを選択してその一覧表をクライアント端末35に対して送信する。クライアント端末35は、その一覧表から順番にプログラムの送信要求を出し、これによってサーバ30から受信したプログラムをクライアントプログラムデータベース38に格納する。要求をしたプログラムの格納が終了すると、(1)の場合の同様にクライアント側バージョン管理テーブルの更新を行う。

【0015】(3)のモードによる処理は、クライアント端末35からサーバ30に対して特定のプログラムの送信要求を送信し、そのプログラムを受信してクライアントプログラムデータベース38に格納する。この場合は、クライアント側バージョン管理テーブルの更新は行わない。

【0016】ステップ15における更新処理が終了すると、ステップ16に移行してベリファイ判断を行う。

【0017】ステップ16におけるベリファイ判断は、ステップ15において更新処理を行ったプログラムが正常であるか否かを調べるための動作を行うか否かをユーザに選択させるためのメッセージを出力して判断を促す。ベリファイを行うと判断した場合はステップ17に移行し、ベリファイを行わないと判断した場合はステップ17をスキップしてステップ18に移行する。

【0018】ステップ17においては、ステップ15において更新処理を行ったプログラムを再度サーバ30から受信し、先にクライアントプログラムデータベース38に格納したプログラムと比較する。この作業は、ステップ15において使用した一覧表に基づいて行われる。ステップ17における処理が終了すると、ステップ18に移行する。

【0019】ステップ18においては、更新した新しい

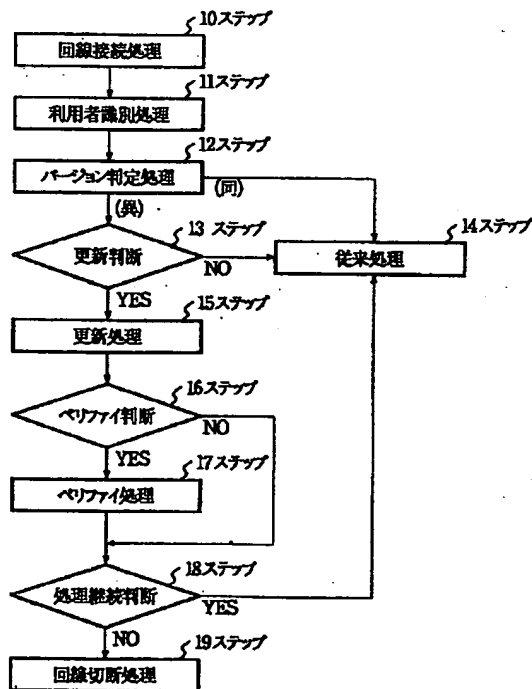
プログラムによって従来処理を行うか否かをユーザに選択させるためのメッセージを出力して判断を促す。従来処理を行うと判断したときはステップ14に移行し、従来処理を行わないと判断したときは、処理を終了させるためにステップ19に移行する。

【0020】ステップ19においては、サーバ30とクライアント端末35との接続を切断するための回線処理を行い、一連の処理を終了する。

【0021】

【発明の効果】以上説明したように、本発明のバージョン管理方法は、サーバからクライアント端末に対してシステムバージョン情報を転送し、システムバージョン情報中のプログラムバージョンがクライアント端末が保有するプログラムバージョンとが一致するか否かの判定し、サーバから送ったプログラムバージョンがクライアント端末が保有するプログラムバージョンと一致しないとき、クライアント端末が保有するプログラムを更新するか否かの判断を行い、更新すると判断した場合はサーバからクライアント端末に対して最新バージョンのプログラムを転送してクライアント端末のプログラムデータベースに格納する更新処理を行うことにより、従来人手で行っていたシステムプログラムのバージョンアップの作業を、自動的に行うことが可能になるという効果があ

【図1】



り、従ってバージョンアップのための作業工数と費用とを節減できるという効果がある。また、ユーザのSEにインストールして貰う必要もなくなるため、信頼性を向上できるという効果もある。

【図面の簡単な説明】

【図1】本発明の一実施例を示すフローチャート図である。

【図2】図1の実施例におけるサーバとクライアント端末との関係を示すブロック図である。

【図3】図1の実施例における更新処理の内容を示す説明図である。

【符号の説明】

10～19 ステップ

30 サーバ

31 バージョン送信部

33・39 バージョン管理データベース

34・38 クライアントプログラムデータベース

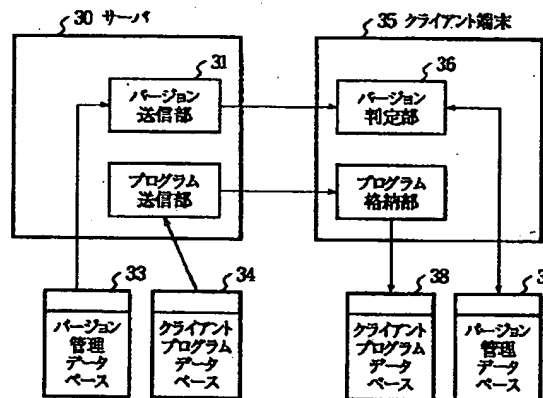
35 クライアント端末

36 バージョン判定部

51 サーバ側バージョン管理テーブル

55a・55b クライアント側バージョン管理テーブル

【図2】



【図3】

